Flexible Architectures for Mandatory Modules: A Reply to Sperber's *Modularity and Relevance*.

Dario Taraborelli
Institut Jean Nicod

Can a massively modular architecture be flexible? This seemingly paradoxical issue is addressed in a recent paper by Sperber (2004), in which the author defends the idea that massively modular systems can smartly manage context-sensitivity. Sperber argues that it is wrong to maintain, with Fodor (1983), that context-dependent tasks require a non-modular, general-purpose architecture and that consequently modules should be limited to rigid peripheral input system. I will shortly review some of the points made by Sperber in defending the idea of flexible modules and I will try to draw some conclusions on the benefits and costs of doing without the idea of mandatoriness.
By way of introduction, let us recall the traditional argument against the hypothesis that the architecture of the human mind might be massively modular. Fodorian modules are characterized as shallow, data-driven, domain-specific, context-independent and mandatory devices. The core feature of a modular system is that it takes only a limited range of inputs and process them *no matter what* according to its internal library of algorithms: it is this relative context-insensitivity and computational rigidity of modules that make them appealing for monitoring low-level information provided by the senses. Fodor (2000) portrays massive modularity as an attempt to save computational psychology in the face of the global sensitivity requirement. Fodorians would argue that since modules are by definition “rigid”, while the human mind is “flexible” (i.e., able to cope with contextual variability), the massive modularity hypothesis is untenable. Sperber's defence of massive modularity consists in challenging this claim, by showing that modularity can well be compatible with context-sensitivity.
The way out the paradox starts from decomposing “flexibility” and “rigidity” into a number of more fine-grained conditions, that allow a module to be adaptive and context-sensitive without losing its very nature. To this aim, the fodorian notion of module - tailored to fit only low-level input devices - must be abandoned: either we stick to fodorian modules and maintain a strong dichotomy between rigid (modular) peripheral mechanisms and flexible (non-modular) central capabilities, or we look for an alternative notion of module, which can be used to account for a number of high-level and context-dependent capabilities without necessarily appealing to Fodor's general purpose device. Sperber justifiably warns against the risk of trivialising the notion of modules by turning them into mere abstract algorithmic "boxes": if the

notion of module is weakened so as to apply to whatever isolable subcomponent of the cognitive system, then the risk is that the modularity hypothesis loses any biological plausibility and becomes a merely abstract characterization of sub-domains of our mental capabilities, which would offer a straight argument to opponents of massive modularity. Preserving modularity without falling into the fodorian trap is then a delicate trade-off requiring a careful revision of the core properties of biologically plausible modules: some of the conditions that characterize fodorian modules must be dropped in order for the modular hypothesis to extend beyond the level of peripheral devices and be applicable to other areas of cognition. The question is then how to soften "rigidity" without losing the benefits of modularity.

Sperber's strategy consists in articulating "flexibility" in two distinct notions, a developmental flexibility (or flexibility "in the long run") and a computational flexibility (or flexibility "in the short run"), and in showing that massive modularity can account for both of them.

The first kind of flexibility can be reached, according to Sperber, through the development of highly adapted sub-modules from innate and domain-specific *module templates*. He provides many an example of pre-wired and domain-specific devices that through exposure to specific application domains or through the pressure of cognitive development can yield dedicated modular subsystems for acquired capabilities like reading, recognizing faces or making arithmetical computations. One might object that allowing the existence of *acquired modules* provide only a partial answer to the flexibility issue and that developmental flexibility is not *per se* a necessary requirement for computational flexibility. Sperber seems to assume that the two notions of flexibility are intimately related, the only difference being the time scale they apply to. We can however conceive a possible taxonomy for articulating long-run and short-run flexibility in which the two notions are contrasted:

<table>
<tr><td colspan="2"></td><td colspan="2">Short-run (computational) flexibility</td></tr>
<tr><td colspan="2"></td><td>Context-insensitive</td><td>Context-sensitive</td></tr>
<tr><td rowspan="2">Long-run (developmental) flexibility</td><td>Rigid</td><td>A) Pre-wired, context-insensitive modules</td><td>B) Pre-wired, context-sensitive modules</td></tr>
<tr><td>Flexible</td><td>C) Adaptive and context-insensitive modules</td><td>D) Adaptive and context-sensitive modules</td></tr>
</table>

Developmentally rigid modules need not be computationally rigid. As an example of pre-wired and context-sensitive modules (B), consider *colour perception*: a built-in module that seems to be able to yield extremely invariant properties (colours) under a high contextual-variability, i.e. global luminance conditions. Conversely, developmentally flexible modules can generate computationally rigid mechanisms, that might indeed benefit from their relative context-insensitivity (C). Replying to the fodorian position, Sperber suggests that massively modular architectures are better described by (D), but we see no reason to exclude (B) and (D) from the set of possible characterizations of massive modular systems.
In particular, the existence of at least four distinct options lays two challenges to Sperber's proposal:

1. How can the distinction between (long-run) rigid and flexible devices be established?
2. Is adaptability in the long run essential for context-sensitivity?

The problem of a developmental rigid/flexible distinction (1.) is extremely delicate, since – as acknowledged by any theory of ontogenetic development – the phenotypic expression of genes (in our case, a module) is co-determined by both the deployment of a genetic programme and the specific environmental domain to which the system is exposed. Sperber captures this distinction by suggesting that modules (or better, *module templates*) can evolve through exposure to domains that do not coincide with their actual domain: a module's *proper domain* of application is fixed by the contingent structure and regularities of the environment to which the cognitive system is exposed during its development. To a certain extent, the idea of modules that do not allow any kind of developmental flexibility is a mere idealization, since even the most rigid devices must undergo a morphogenesis that requires exposure to a specific environment. One could even argue that talking of actual domains for *module templates* without referring to specific environmental contexts can result in mere abstractions: what is the actual domain from which acquired modular capacities like numeric cognition or face recognition evolved? Lacking a characterization of what makes module templates flexible and other modular devices rigid from the point of view of cognitive development, it is difficult to see how a clear-cut distinction between developmental rigidity and flexibility can be drawn.
As for 2., Sperber speculates that "*a system that is flexible is in a better position to exhibit context-sensitivity in the short-run*". Although plausible, this seems hardly be the case for a number of modules that, permeable to cognitive development, exploit their computational rigidity (i.e. context-insensitivity) as a key-feature. It might be profitable in many situations for a system to dispose of a set of highly adapted modular devices that do not display computational flexibility and provide default

solutions independently of contextual information. Literature on cognitive and perceptual biases is rich of examples of how modules that are flexible from the viewpoint of long-run learning, actually use their computational rigidity to provide optimal solutions for specific kinds of task.

Defendants of a fodorian view might still agree with this articulation of developmental and computational flexibility, without accepting the existence of computationally rigid mechanisms at higher levels of cognitive architecture. What is at stake, in this case, is not much the fact that a modular architecture might undergo a learning process and adapt 'rigid' resources to the solution of problems typical of acquired competences. Flexibility, in its stronger meaning, is not a matter of adaptability across cognitive development, but a problem of processing efficiency in complex situations. Flexibility means being able to tame context-variability in real-world tasks.

Fodorians would claim that modules cannot be flexible because they react *rigidly* in presence of particular classes of stimuli belonging to the module's domain of application. Sperber's answer to this objection is that flexibility *can* be attained if modules are not taken to work in a strictly mandatory fashion: the modularity hypothesis is compatible with context-sensitivity if we drop a core property of fodorian modules, i.e.: *mandatoriness*. "*If one takes for granted that modularity implies mandatoriness* - he observes - *then one should reject the massive modularity hypothesis*".

What should be mandatoriness dropped instead of other core properties of fodorian modules like encapsulation, domain-specificity or hardwiredness ? The reason is that, according to Sperber, context-sensitivity is a matter of having a procedure not running its full course in spite of the availability of the appropriate input. "*Humans can discriminate tens of thousands of categories in their environment, very few of which trigger automatic behavioural responses. At any one moment, humans are monitoring their environment through all their senses and establish perceptual contact with a great many potential inputs for further processing (...) Cognitive efficiency is a matter of investing effort in processing the right inputs*".

If a system is mandatory, it cannot be stopped from producing its output when the input conditions are present. Both Sperber and defendants of a fodorian view agree that such a mechanism can hardly solve tasks in which the system is required to deal with context-dependence problems.

Let us illustrate this point with an example. Imagine that the same property A (like the presence of a quivering black dot in the visual field) can either signal to a frog the presence of an edible bug (B) in a dry environment or the presence of a poisonous insect (C) near by a swamp. Being able to distinguish between an edible and poisonous insect requires being able to distinguish between two environmental contexts. A mandatory device (like a *quivering-black-dot detector*) triggering the same grasping mechanism in both cases is an example of 'rigid' device, unable to deal with context-variability. Generalizing the problem to cognitively complex situations, one can easily mention cases in which context-sensitivity becomes computationally overwhelming. The fodorian answer to the necessity of flexible capabilities to cope with context-variability is to reject a massively modular architecture in favour of general-purpose high-level devices; Sperber's answer, on the contrary, is to postulate at any level of the system's architecture the existence of flexible modules that do not work in a mandatory fashion when the appropriate stimulus is available.

Taming context-variability is demanded in Sperber's account to the allocation of energetic resources to modules. If a module is exposed to the appropriate kind of input but is not allocated sufficient resources to trigger its internal procedure, it won't produce any output. In the example of inattentional blindness mentioned by Sperber, not being aware of the gorilla appearing on the screen while the subject is attentionally involved in a side-task is a matter of not having allotted sufficient resources to the *gorilla detector module*.

The advantages of such an hypothesis, on the one hand, are clear: dedicated computational devices can populate any level of the cognitive architecture while there is no need to postulate a massive number of inhibitory mechanisms to prevent all undesired stimulus properties from being processed, nor the existence of a meta-module for evaluating what are relevant properties to be processed: it is enough to postulate non-cognitive mechanisms that allow the system to optimize the distribution of resources, as a function of previously learned cost/effort ratios. In this sense, "*different modules may be more or less mobilised in a way that reflects their general contribution to relevance*".

It should be stressed, on the other hand, that what gets lost with the idea that modules can be flexibly mobilised is precisely one of the core features of fodorian modules, i.e. that modules yield their output *no matter what*. One of the reasons why Sperber needs modules to be non-mandatory is that he implicitly assumes that, if this were not the case, a sequence of modules like the following:

SENSORY INPUT ⇒ [Quivering dots detector] ⇒ [Edible stuff detector] ⇒ [Motor scheme selector] ⇒ BEHAVIOURAL OUTPUT

would always yield the behavioural output when the appropriate sensory input is presented. This assumption raises a number of interesting problems:

1. First, saying that if we allow mandatoriness then we must accept the above conclusion is unjustified. The idea that mandatoriness of *single modules* in the sequence entails that *the whole sequence is mandatory* is too hasty a conclusion that stretches mandatoriness outside the scope of single modules. Sperber's

strategy consists in trivializing the idea of mandatory mechanisms, by suggesting that mandatory modules necessarily lead to a mandatory architecture, whereas the very idea of resource allocation he proposes seems rather compatible with the fact that modules can be mandatory devices mobilized by a flexible architecture. Resource allocation operates outside the scope of modules: it prevents, for instance, that an edible-stuff detector be triggered by the output of a quivering-black-dot detector, if more energy is directed to a concurrent process. In no way it affects the internal functioning of a module. One might reformulate this idea and say that active modules are mandatory even if their activation depends on the amount of allocated resources; inactive modules are simply *inactive*.

2. One could be unsatisfied with the above argument and object that redefining flexibility as a property of architecture instead of as a property of modules is nothing more than a terminological trick. To see why this could not be the case, consider the following alternative characterization of Sperber's resource allocation idea. What is the benefit of having a resource allocation mechanism operate on the input of a module rather than on its *output*? Take the case of the inattentional gorilla. Sperber correctly argues that the reason why the gorilla is not consciously perceived is that (limited) attentional resources are already allocated to more relevant tasks. Sperber assumes that if the gorilla detector were mobilised one would be forced to see a gorilla, hence the gorilla-detector must have been somehow prevented from working. Again, this is a trivialization strategy that can be avoided as soon as we consider *who* is seeing the gorilla and what is meant by *seeing*. Where is the output of the gorilla-detector module to be evaluated? We might alternatively refer to a personal-level conscious report or to the activation of some further sub-personal device. We agree with Sperber's conclusion only if we stick to the first option. It looks extremely plausible, though, that – even if the gorilla was not consciously reported – some kind of implicit processing (that priming tests could reveal) might nonetheless have occurred. The fact that the Gorilla is not consciously reported does not enable one to say whether it is at the input of the gorilla detector that the process is inhibited or rather at the output. For example, if the behavioural response had been different from an introspective report (first person phenomenology is seldom reliable for understanding underlying cognitive mechanisms), there might be chances for some kind of unaware "gorilla processing" to show up. Under this reading, the description of the gorilla experiment seems at least as compatible with Sperber's proposal as with my suggestion: the module was activated, but its output was not given the necessary resources to go further in the processing sequence, in particular to mechanisms mediating conscious access. Gorilla-detector modules *are* mandatory, while architectural constraints and energy allocation can make their output more

or less likely to subserve further processing. There seems to be, in the end, no convincing reason for rejecting the alternative idea that, instead of weakening a module's mandatoriness (i.e., instead of making modules more or less activated according to allocated resources), modules could be seen as producing their output *no matter what*. It is the relevance and congruence of their output that establishes whether some information can go further in the processing sequence. To rephrase Sperber, "*cognitive efficiency*" could be "*a matter of investing effort in selecting the right outputs*".

3. What is interesting, then, in shifting relevance mechanisms from the input to the output of modules? I submit that the interest of applying the modularity hypothesis to different levels of cognitive architecture is inseparable from the idea that modules are mandatory. Making modules flexibly activated as a function of input relevance conditions weakens one of their core feature, i.e. their *no-matter-what* functioning, thus offering to opponents of the massive modularity idea an easy chance to trivialize the notion of module. Domain-specific systems that are flexibly activated as a function of their input's relevance for the task's context look dangerously similar to general-purpose non-modular mechanisms like those postulated by fodorians. One might justifiably ask: are non-mandatory modules still "modules"? Keeping the idea of a resource allocation strategy without renouncing to computational rigidity of modules seems – at least as a general option – a better strategy for defending massive modularity.

It might be argued that, to a large extent, the problem of the *locus of activity* of relevance selection mechanism is an empirical question. It might be undecidable on theoretical grounds (and maybe purely speculative) whether A) modules are simply not activated when they are not allocated sufficient resources or if B) they are activated but their output is unable to undergo further processing when it is overridden by the output of more relevant processes.
Still, it seems that avoiding to drop mandatoriness, or keeping a notion of mandatoriness compatible with a flexible architecture, might help the cause of massive modularity. What is gained by allowing modules to be mandatory is the fact that one can account for the achievement of a particular task as the result of a competition between *activated* modules. We agree with Sperber that processes are competing for resource allocation, but shifting the competition on the output of the process instead of on input selection has the interesting advantage of keeping flexibility without renouncing to an essential feature of modules. Filtering relevance at the output of a process is compatible with the idea that a module yielding irrelevant information in a specific context will be more and more often overridden in the long run by modules producing more relevant information. A module whose output loses causal effects

could then be described (pace Sperber) as a device that is not *mobilised*: the only difference is that it is not mobilised because using its output instead of other modules' output in the same context has proved too costly/risky for the cognitive system in its past experience. This shows incidentally that, if one abandons the strategy of trivializing mandatoriness, the present proposal is to a large degree compatible with Sperber's model: if the output of a process is not allocated enough resources to feed another module, then *a fortiori* the second module is not mobilised, which is exactly the idea that Sperber is defending.

The general conclusion of this analysis supports the idea that massive modularity can smartly cope with problems of context-sensitivity. The role of relevance selection mechanisms and energy allocation proposed by Sperber (2004) is crucial for understanding how a massively modular architecture can be computationally flexible. However, weakening the notion of modularity so as to make modules non-mandatory can seriously threaten the claim that such an architecture might still be considered *modular*. By suggesting that relevance mechanisms operate on competing output of modules, and that mandatory modules do not entail a mandatory architecture, I have showed how a non-trivial notion of module can be defended without renouncing to computational flexibility.

References

Fodor, J. A. (1983). *The modularity of mind: an essay on faculty psychology*. Cambridge, Mass.: MIT Press.

Fodor, J. A. (2001). *The Mind Doesn't Work That Way*. Cambridge, Mass.: MIT Press.

Sperber, D. (2004). Modularity and relevance: How can a massively modular mind be flexible and context-sensitive ? in P. Carruthers, S. Laurence & S.Stich (Eds.), *The Innate Mind: Structure and Content.*